

PCT

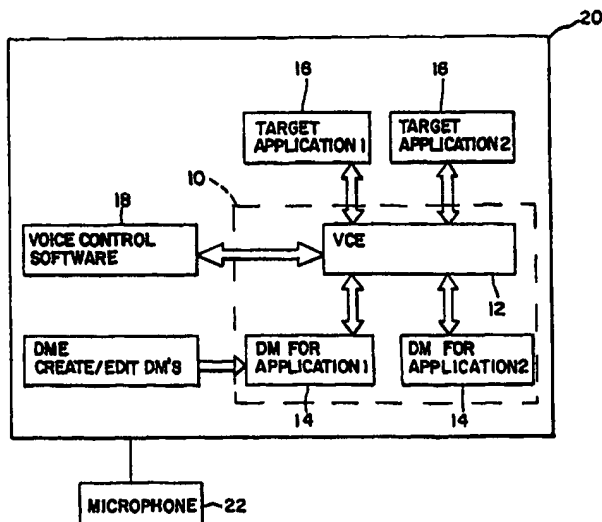
WORLD INTELLECTUAL PROPERTY ORGANIZATION  
International Bureau



INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification <sup>6</sup> : <b>G10L 5/00</b>		A1	(11) International Publication Number: <b>WO 99/05671</b>
			(43) International Publication Date: 4 February 1999 (04.02.99)
(21) International Application Number: PCT/US98/15213		(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: 23 July 1998 (23.07.98)			
(30) Priority Data: 60/053,621 24 July 1997 (24.07.97) US			
(71) Applicant (for all designated States except US): KNOWLES ELECTRONICS, INC. [US/US]; 1511 Maplewood Drive, Itasca, IL 60143 (US).			
(72) Inventor; and			
(75) Inventor/Applicant (for US only): GARRATT, Sean, R. [US/US]; 316 Milan Court, Schaumburg, IL 60193 (US).			
(74) Agent: MORNEAULT, Monique, A.; Wallenstein & Wagner, Ltd., 311 South Wacker Drive - 5300, Chicago, IL 60606 (US).			
		Published With international search report. Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.	

(54) Title: UNIVERSAL VOICE OPERATED COMMAND AND CONTROL ENGINE



(57) Abstract

The present invention is a system for controlling graphical user interface by voice commands. The present invention constitutes a means for receiving issued voice commands from a standard voice recognition system (18), a means for monitoring the state of a target application (16), a means for determining active voice commands from the state of the target application (12), a means for determining whether issued voice command is an active voice command, a means for associating each active voice command with a block of script code data (14), a means for issuing the block of script code data associated with the issued voice command to the graphical user interface when the issued voice command is determined to be an active voice command.

AL

**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakhstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

**UNIVERSAL VOICE OPERATED COMMAND AND  
CONTROL ENGINE**

5 This is application claims the priority of U.S. Provisional Application  
Serial Number 60/053,621 filed July 24, 1997.

**Technical Field**

10 The present invention relates generally to software for a  
computer system which provides voice operated control of a graphical  
user interface such as a Microsoft WINDOWS based environment.  
More specifically, the present invention relates to a system which takes  
standard Microsoft Speech Application Programming Interface  
compliant commands from a speech recognition software package and  
directs them to a target application in order to control that target  
15 application.

**Background of the Invention**

20 Generally, there are two applications of speech recognition  
technology in the computer speech recognition art: 1) dictation  
applications and 2) command and control applications. Dictation  
applications recognize the speech of a computer user in order to  
reproduce those words in a computer software application, such as a  
word processor. Therefore, a user may dictate a letter to the computer  
as opposed to manually typing it out. Command and control  
25 applications, on the other hand, recognize the speech of a computer  
user in order to operate the computer itself. Therefore, a user may  
issue commands to a computer, such as to execute a program, save a  
file, or to change the font of the letter being dictated, rather than  
manually use a keyboard or mouse to issue the command. Speech  
30 recognition used to perform either of these two tasks can greatly  
increase the productivity of a computer user.

Voice recognition software, such as IBM's VOICETYPE or

Dragon Systems' NATURALLY SPEAKING, is well known in the art. Generally, voice recognition software, in conjunction with a computer system, converts an analog voice signal into digital data capable of interacting with a computer system. Software also exists which enables computer users to make event calls to the operating system that simulate actions by well-known computer input devices such as a mouse or a keyboard, thereby allowing computer users to interact with the graphical user interface via voice control.

However, no computer software application exists which allows a user to create new voice commands or change the behavior of current voice commands by using standard scripting languages. The ability to use standard scripting languages allows the creation of local and global variables which can be shared by different voice commands in order to create more comprehensive voice commands.

Additionally, no computer software application exists which will: 1) continually monitor an operating system or software application state, and its subwindow state, to be voice controlled for a listing of voice commands which may be validly issued and 2) dynamically maintain the listing of voice commands as the state of the operating system or software to be controlled changes. The tracking of the state of the operating system or software application to be controlled also allows commands to be issued based on the state of the operating system or software application.

#### Summary of the Invention

The present invention is directed to computer software for converting spoken commands into commands capable of directing graphical user interfaces of any computer application for a particular computer system.

Specifically, the present invention comprises a means for receiving issued voice commands from a standard voice recognition system, a means for monitoring the state of a target application, a

means for determining active voice commands from the state of the target application, a means for determining whether the issued voice command is an active voice command, a means for associating each active voice command with a block of script code data, and a means for  
5 issuing the block of script code data associated with the issued voice command to the graphical user interface when the issued voice command is determined to be an active voice command.

Further, the present invention will be an improvement over current technology based on its methodology of processing events  
10 within the accompanying operating system and its ability to adapt to virtually any Microsoft WINDOWS based application. The present invention accomplishes this by using industry-standard scripting languages. Additionally, the present invention monitors target applications to determine which voice commands may be validly issued  
15 based on the target application state. The present invention does this in order to prevent invalid commands from being issued to the target application regardless of whether the original application was intended to be controlled through voice commands.

Other advantages and aspects of the present invention will  
20 become apparent upon reading the following description of the drawings and detailed description of the invention.

#### Brief Description of the Drawings

Figure 1 is a chart showing the interrelation of the present  
25 invention with external components;

Figure 2 is a chart displaying the internal operation of the invention;

Figure 3 is a plan view of a desktop icon according to the present invention;

30 Figure 4 is a plan view of a configuration dialog according to the present invention;

Figure 5 is a plan view of a voice command dialog according to the present invention;

Figure 6 is a chart showing the contents of a directive module data file according to the present invention.

5           Figure 7 is a plan view of a directive module editor according to the present invention;

#### Detailed Description

10           While this invention is susceptible of embodiment in many different forms, there is shown in the drawings and will herein be described in detail a preferred embodiment of the invention with the understanding that the present disclosure is to be considered as an example of the principles of the present invention and is not intended to limit the broad aspect of the invention to the embodiments illustrated.

15           Referring to Figure 1, generally, a Universal Voice Operated Command and Control Engine 10 consists of two primary components: a Voice Control Engine ("VCE") 12 and a Directive Module ("DM") 14. In order for a user to control a target application 16 by a voice command, the user issues the voice command which is then passed to  
20           Voice Control software 18 running on a computer 20 with a microphone 22 connected. The Voice Control software 18 then recognizes the speech pattern and passes the result to the Universal Voice Operated Command and Control Engine 10. In the present invention, the Voice Control software 18 can be any commercially  
25           available voice recognition software which adheres to Microsoft's Speech Application Programming Interface ("SAPI") standards such as Dragon Systems' NATURALLY SPEAKING and IBM's VOICETYPE. SAPI compliance means that a users choice of speech recognition  
30           software is transparent to the VCE 12, providing the vendor of the speech recognition software adheres properly to the SAPI standards.

As will be explained in further detail below, the VCE 12 then

determines if the command is proper by querying the target application 16 to determine its present state. If the voice command is proper, the Universal Voice Operated Command and Control Engine 10 executes a section of computer code contained within the DM 14 corresponding the particular target application 16 in order to control the target application 16 for executing the voice command. The computer code that is executed is preferably written in Visual Basic for Applications ("VBA") code, although other scripting languages such as Javascript can be used.

#### Voice Control Engine

Referring to Figure 2, the VCE 12 is the run-time software for receiving information from the Voice Control software 18, dispatching synthetic keyboard and mouse messages to the Operating System ("OS") or target application 16, and monitoring the current target application state 16.

Specifically, the VCE 12 interfaces with the Voice Recognition software 18 through a standard SAPI interface 24 with a SAPI controller 26. The SAPI controller 26 comprises interface code to initialize a conversation with the SAPI interface 24, to monitor a success state of the SAPI interface 24, and produce notification callbacks from the SAPI interface 24. The success state of the SAPI interface 24 indicates whether a command transmitted through the SAPI interface 24 was successful. The SAPI controller also receives updated active voice commands from an Active Voice Command Updater 28 as the state of the target application 16 changes. Once a conversation is initialized by the SAPI controller 26, the SAPI controller 26 receives commands from the Voice Recognition software 18 and compares the voice command received from the SAPI interface to the commands provided by the Active Voice Command Updater 28.

If the received voice command matches a command provided by the Active Voice Command Updater 28, then it is a presently valid

voice command, and the SAPI controller 26 passes the command to a Recognized Voice Command Handler 30. The Recognized Voice Command Handler 30 then forwards the command to a DM Parser and Indexer 32 if the voice command is "simple" and contains no variable data. However, if the voice command contains variable data, such as "Set font size to {number}," the Recognized Voice Command Handler 30 preprocesses the voice command before passing it to the DM Parser and Indexer 32. If the received voice command does not match a command provided by the Active Voice Command Updater 28, then it is a presently not valid voice command, and the SAPI controller 26 can either take no action or return a message to the SAPI Interface 24 that the command was not valid.

The DM Parser and Indexer 32 then takes the voice command and, using the DM 14 corresponding to the active application, determines the section of scripting code corresponding to the voice command received, and issues the scripting code to the Message Dispatcher 34. The Message Dispatcher 34 then issues pseudo mouse and/or keyboard messages to the Operating System or target application 16. The Message Dispatcher 34 accomplishes this through standard WIN32 API calls such as SendMessage() or PostMessage().

The DM Parser and Indexer 32 also passes the list of the active voice commands to the Active Voice Command Updater 28 when the DM Parser and Indexer receives an updated list of active voice commands from a Current Application State Monitor 36. The Current Application State Monitor 36 continuously polls the active target program 16 to determine its state and maintains the list of active voice commands. The Current Application State Monitor 36 does this by determining which dialog and form is open in the target application 16 at the present time and sending the commands that may be validly issued to the DM Parser and Indexer 32.

A DM Loader/Container 38 holds DMs 14 for parsing and



indexing by the DM Parser and Indexer 32, and loads, or unloads, DMs 14 whenever the DM Loader/Container 38 recognizes a new target application 16 has loaded, or unloaded, for which a DM 14 is available. The DM Loader/Container 38 retrieves the DMs 14 from an electronic media storage 40, such as a hard drive, of the computer 20.

As has been described, the VCE 12 has no controls which are visible to the user of the computer 20. However, as in the Windows 95 platform, preferably a small icon 42 is visible in the system tray 44, as shown in Figure 3, to indicate to the user that the VCE 12 has been loaded. Additionally, the icon may be clicked with the left mouse button to show a small menu of possible command options, such as: an option for showing a User Interface 46 for displaying the list of active commands, a stop voice commands option, a start voice commands option, an option to open a User-Definable Variables dialog 48 (Figure 4), an option for online help, and an option to unload the VCE 12.

The User-Definable Variables dialog 48, as shown in Figure 4, includes an option to require confirmation of every voice command, a user definable confirmation statement for confirmation, an option for showing the list of active commands dialog every time the VCE 12 initially loads, an option to open the Voice Recognition software's 18 recognition parameters window, an option to run an SAPI microphone variable adjustment dialog, an option to set the maker of the Voice Recognition software 18, and a listing of installed DMs 14 and means for disabling any one of the installed DMs 14.

Finally as explained above, the VCE's 12 User Interface 46 dialog, as shown in Figure 5, contains a list of active commands for which voice commands may be validly issued. The User Interface 122 dialog also has provisions to select an active command with a mouse or keyboard in order to issue the command, rather than issue it by voice. Through a pull-down menu and/or toolbar there is provided: an option to sort the list of commands alphabetically or by most used commands

first, an option to keep the dialog visible even when the dialog is not the active, a smart locate function for the dialog, and an option to hide the window. In the smart locate function, the dialog attempts to locate itself to a portion of the screen where it does not block the user's view.

5     Directive Module Format

As explained above, each DM 14 is a separate data file which contains information about a respective target application. The layout of a DM 14 is shown in Figure 6. The DM 14 contains Header data, Form Template data, and Script Code data. The Header data contains  
10     information such as the creation date and version of the DM 14, data about the target application 16, such as target application name, version, target application executable file location, etc., a pointer to the beginning location of the Form Template Data, and a pointer to the beginning location of the Script Code Data.

15     Within the Form Template data, information about an application's dialogs and forms is stored for use by the VBA script code, such as locations of command buttons, contents of list boxes, etc. within each individual form or dialog. Within the Script Code section is the scripting code to perform the voice commands within the target  
20     application 16. The script code is passed to the target application through standard WIN32 API calls, such as SendMessage() or PostMessage(). Additionally, the most preferred scripting language is VBA, however, Javascript, or any other scripting language, could be implemented.

25     Directive Module Editor

Additionally, as shown in Figure 7, there is shown a Directive Module Editor ("DME") 100 application. The DME 100 is not needed in order to implement the function of the VCE 12, but is used as a development tool in order to edit DMs 14. The DME 100 includes the  
30     ability to load DMs 14 for the purpose of adding or removing voice commands to a DM 14, and the scripting code associated with that

command. The DME 100 preferably comprises: a command window 102 in which available commands are listed, a form template window 104 in which the form or dialog box of the target application to be controlled is shown, and a scripting code window 106. A user may then edit DMs 100 to include more complex commands or sets of commands to add more functionality to a single voice command.

While the specific embodiments have been illustrated and described, numerous modifications come to mind without significantly departing from the spirit of the invention and the scope of protection is only limited by the scope of the accompanying Claims.

## I CLAIM:

1. A system for controlling a graphical user interface comprising:  
means for receiving a voice command;  
5 means for determining active voice commands;  
means for determining whether the received voice command is an active voice command; and,  
means for executing a received voice command if the received voice command is an active voice command.

2. The system of claim 1, wherein the means for executing a received voice command comprises:  
means for associating voice commands with a block of script code data; and,  
15 means for issuing the block of script code data associated with the received voice command to the graphical user interface when the received voice command is determined to be an active voice command.

3. The system of claim 2 wherein:  
20 the means for determining active voice commands includes the ability to determine variable data contained within the target application; and,  
the means for issuing the block of script code data includes the ability to incorporate the variable data within the script code data.

4. The system of claim 1, wherein the means for receiving issued voice commands is a standard voice recognition system.

5. The system of claim 4, wherein standard voice recognition system is a SAPI compliant voice recognition system.

6. The system of claim 1, wherein the means for determining active voice commands comprises a means for monitoring the state of a target application.

5           7.     A system for controlling a graphical user interface comprising:

a directive module comprising script code data;

a voice control engine comprising:

means for receiving a voice command;

10           means for determining active voice commands;

means for determining whether the received voice command is an active voice command;

means for executing a received voice command if the received voice command is an active voice command.

15           8. The system of claim 7, wherein the means for executing a received voice command comprises:

means for associating voice commands with the block of script code data; and,

20           means for issuing the block of script code data associated with the received voice command to the graphical user interface when the received voice command is determined to be an active voice command.

9. The system of claim 8, wherein:

25           the means for determining active voice commands includes the ability to determine variable data contained within the target application; and,

the means for issuing the block of script code data includes the ability to incorporate the variable data within the script code data.

30           10. The system of claim 7, wherein the means for receiving

issued voice commands is a standard voice recognition system.

11. The system of claim 10, wherein the standard voice recognition system is a SAPI compliant voice recognition system.

5

12. The system of claim 7, wherein the means for determining active voice commands comprises:

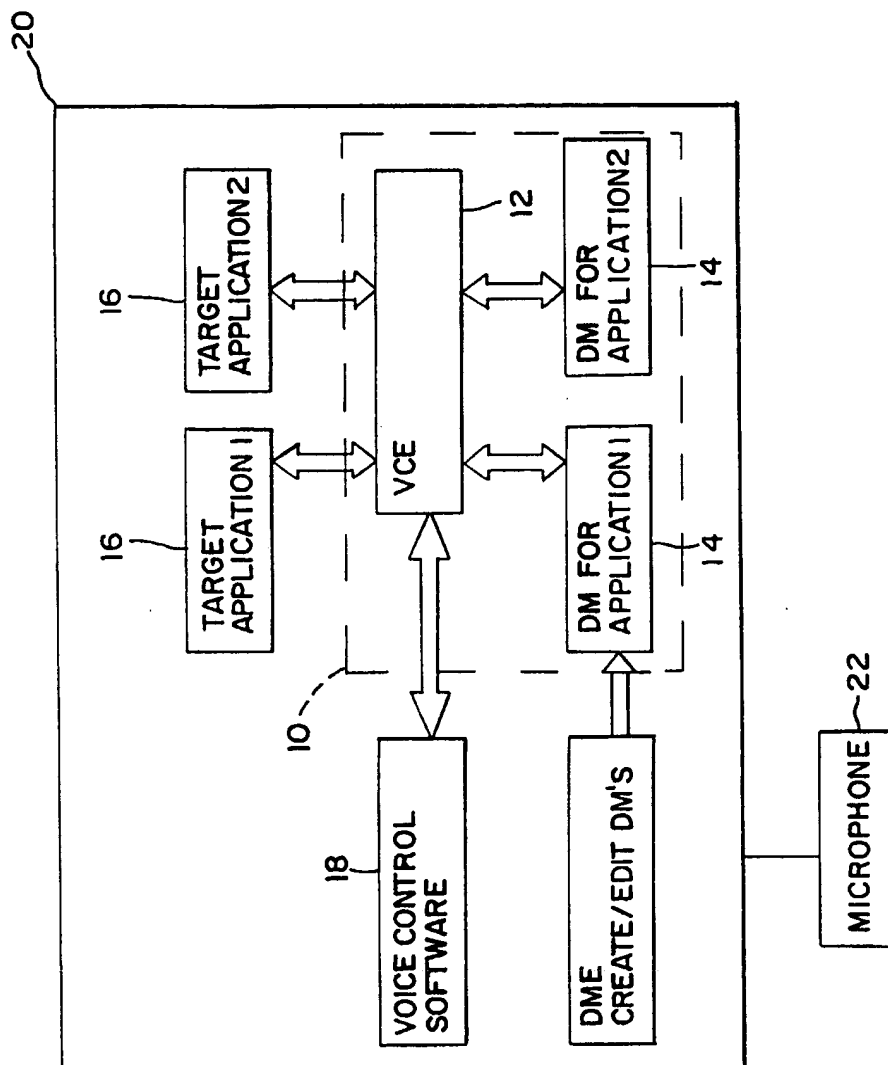
means for monitoring the state of a target application; and,

means for comparing the state of a target application to the

10

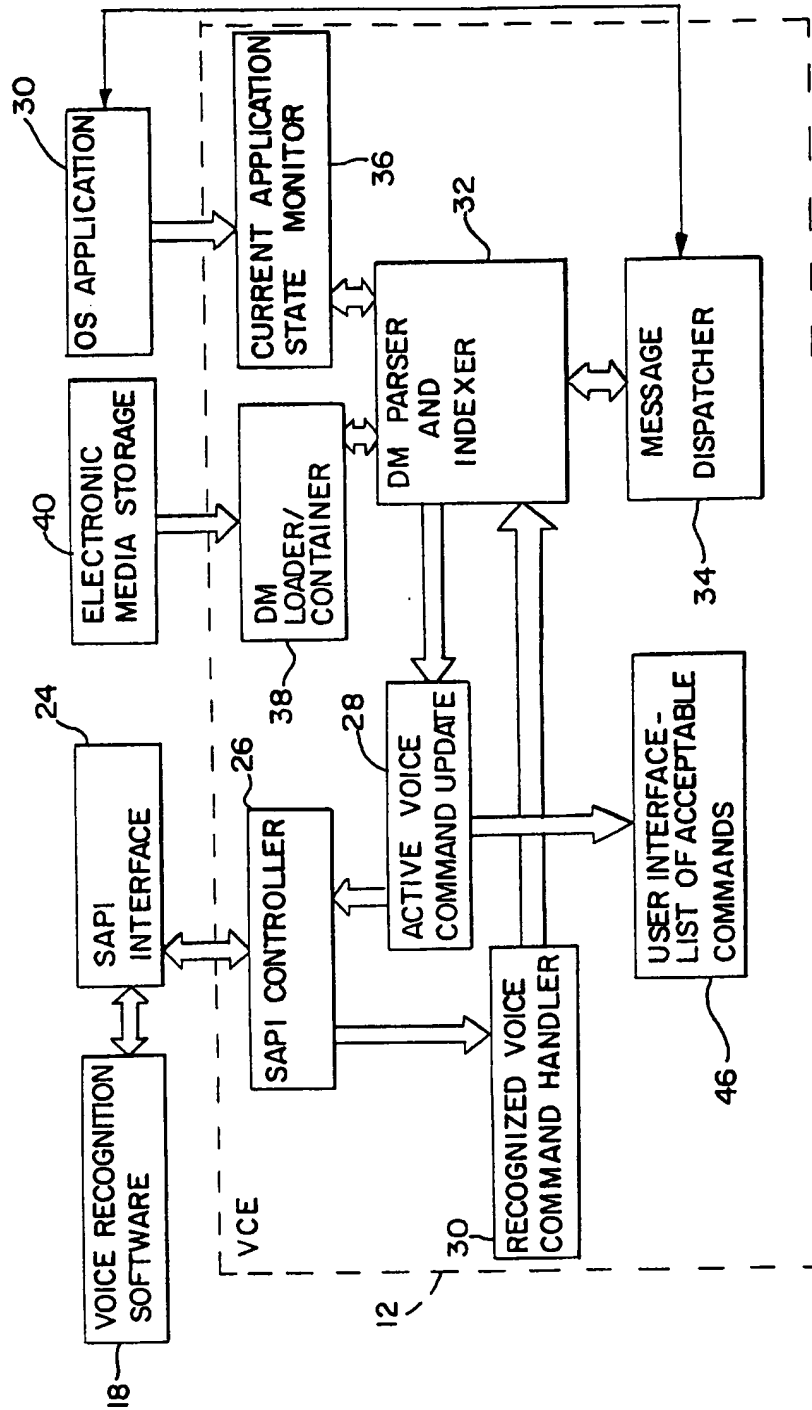
script code data within the directive module to obtain the active voice commands.

1 / 5



2 / 5

FIG. 2





3 / 5

FIG. 3

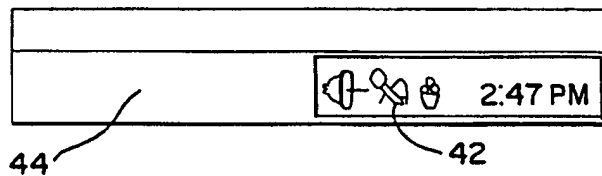
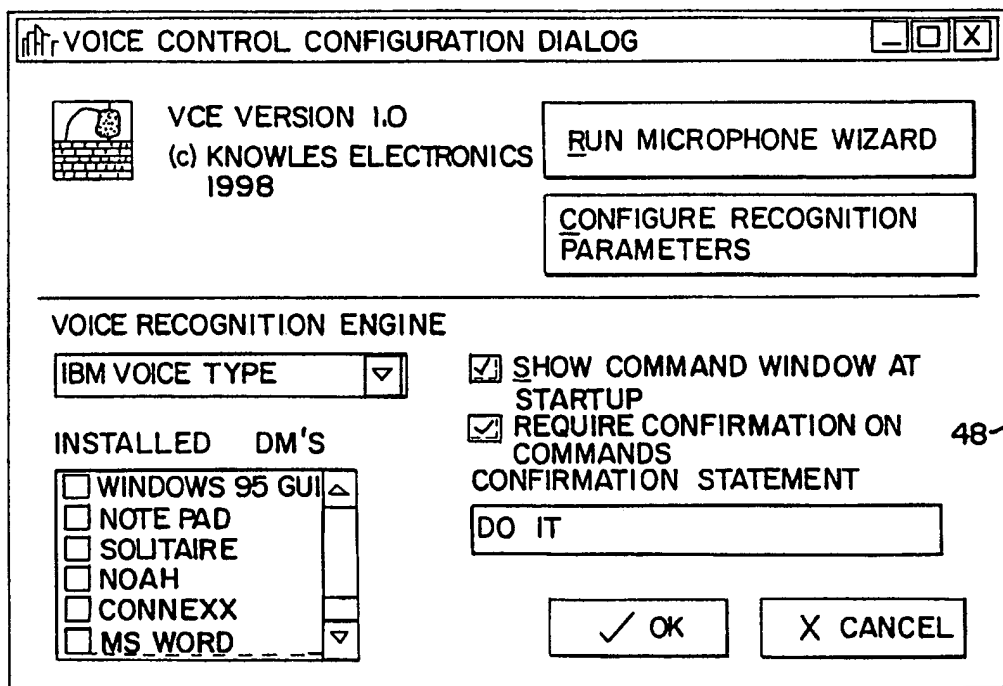


FIG. 4



4 / 5

FIG. 5

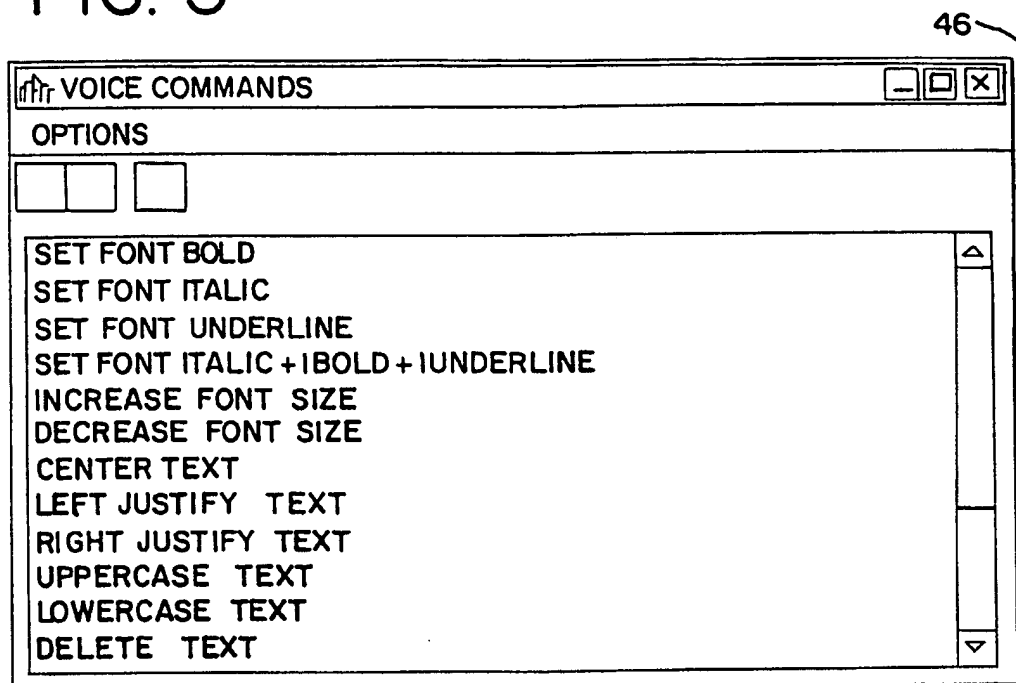
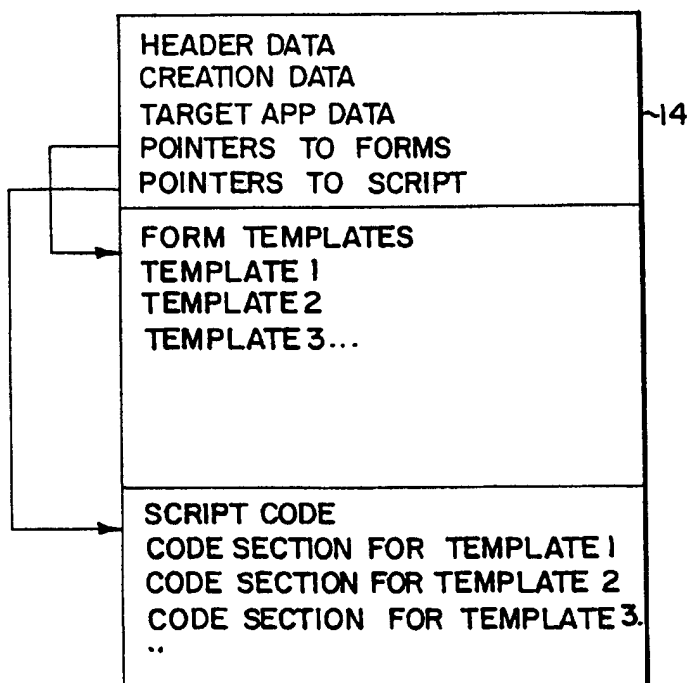


FIG. 6



5 / 5

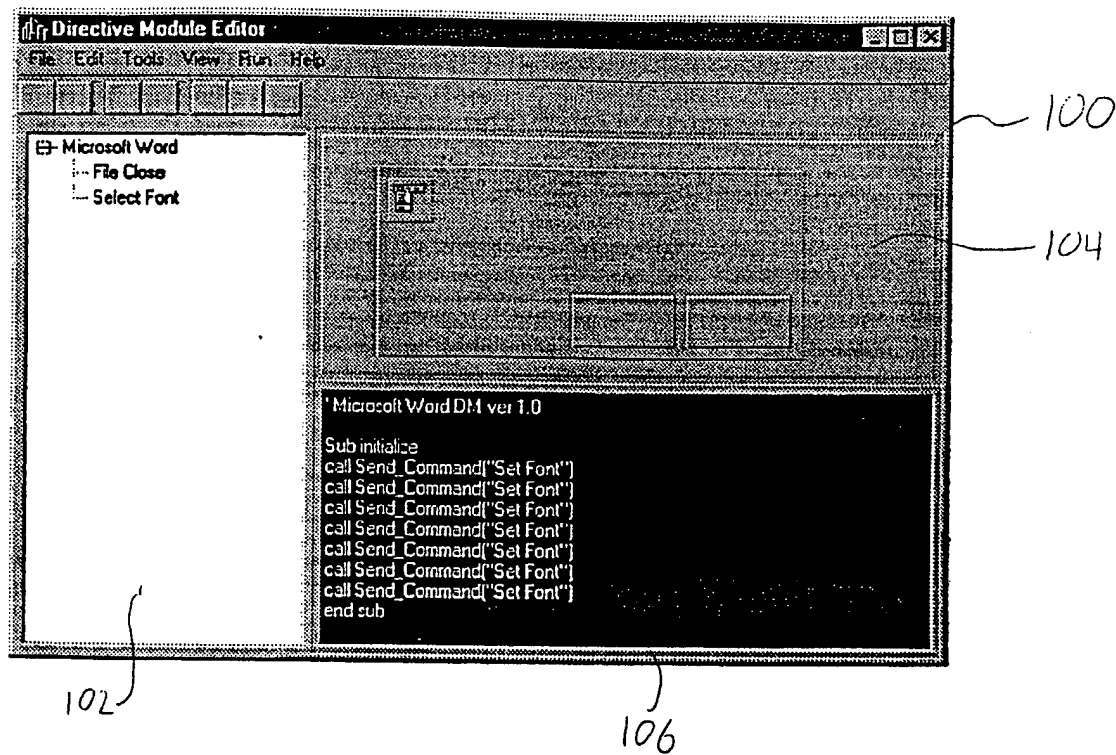


FIG. 7

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US98/15213

**A. CLASSIFICATION OF SUBJECT MATTER**

IPC(6) :G10L 5/00  
US CL :704/275,270

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 704/275,270,251,241,231; 345/340,326

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched  
none

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)  
APS, MAYA, smart patent workbench

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 4,513,189 A (UEDA ET AL.) 23 April 1985, abstract, Col.1, lines 65-68, Col.2, lines 1-13.	1-12
X	US 4,776,016 A (HANSEN) 04 October 1988, Figs. 5a, 5b, Col.4, lines 1-61.	1-12
X	US 5,231,670 A (GOLDHOR ET AL.) 27 July 1993, title, abstract, Fig.1, Col.2, lines 28-33, 40-55.	1-12
X	US 5,267,323 A (KIMURA) 30 November 1993, abstract, Fig.3, Col.4, lines 10-30.	1-12
X	US 5,345,538 A (NARAYANNAN ET AL.) 06 September 1994, abstract	1,7

☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:	*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
*A* document defining the general state of the art which is not considered to be of particular relevance	*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
*B* earlier document published on or after the international filing date	*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
*L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	*G* document member of the same patent family
*O* document referring to an oral disclosure, use, exhibition or other means	
*P* document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search 17 SEPTEMBER 1998	Date of mailing of the international search report 31 DEC 1998
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer VIJAY CHAWAN <i>Joni Hill</i> Telephone No. (703) 308-3900

# INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US98/15213

## C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	US 5,659,665 A (WHELPLEY, JR.) 19 August 1997, abstract	1,7